Journal of Computational Mathematics, Vol.23, No.6, 2005, 619–634.

A FEASIBLE DIRECTION ALGORITHM WITHOUT LINE SEARCH FOR SOLVING MAX-BISECTION PROBLEMS *1)

Feng-min Xu Cheng-xian Xu Hong-gang Xue
(*Department of Mathematics of Faculty of Science, Xi'an Jiaotong University, Xi'an, 710049, China*)

Abstract

This paper concerns the solution of the NP-hard max-bisection problems. NCP functions are employed to convert max-bisection problems into continuous nonlinear programming problems. Solving the resulting continuous nonlinear programming problem generates a solution that gives an upper bound on the optimal value of the max-bisection problem. From the solution, the greedy strategy is used to generate a satisfactory approximate solution of the max-bisection problem. A feasible direction method without line searches is proposed to solve the resulting continuous nonlinear programming, and the convergence of the algorithm to KKT point of the resulting problem is proved. Numerical experiments and comparisons on well-known test problems, and on randomly generated test problems show that the proposed method is robust, and very efficient.

Mathematics subject classification: 90C27
Key words: Max-Bisection problem, Feasible direction algorithm, NCP function, Convergence.

1. Introduction

This paper concerns the solution of the max-bisection problem for a given undirected graph $G = (V, E)$ with $V = \{1, 2, \cdots, n\}$ the node set, E the edge set, and n is even. Let $W = (w_{ij})_{n\times n}$ be the symmetric weight matrix with $w_{ij} > 0$ if $(i, j) \in E$ and $w_{ij} = 0$ if $(i, j) \notin E$. The max-bisection problem is to partition the node set V into two subsets S and $V \backslash S$ having equal cardinality such that the sum $\sum_{i\in S,\ j\in V\backslash S} w_{ij}$ is maximized. The problem can be formulated by assigning each node a binary variable x_j

$$(MB):\begin{cases} MB(S) = \text{Max } \ \frac{1}{4}\sum\limits_{i,j} w_{ij}(1 - x_i x_j) \\ s.t.\ \ e^T x = 0, \\ \quad x_j^2 = 1, \quad j = 1, \cdots, n, \end{cases}$$

where $e \in R^n$ is the column vector of all ones. The constraint $x_j^2 = 1$ implies that x_j takes either 1 or -1, so that we will have either $S = \{j|x_j = 1\}$ or $S = \{j|x_j = -1\}$. The constraint $e^T x = 0$ ensures $|S| = |V \setminus S|$.

The max-bisection problem is NP-hard [1], and has wide applications in real world. Approximation algorithms are available, and polynomial time approximation schemes exist for the problem over dense graphs [3] and over planar graphs [4]. Frieze and Jerrum[6] extended Goemans-Willamson approach [5] to max-bisection problems, giving a randomized 0.651 approximation algorithm for the maximum weight bisection problem. Ye [7] improved the performance ratio of the algorithm to 0.6993 by combining the Frieze-Jerrum approach with some

* Received September 21, 2004; Final revised May 6, 2005
1) This work is supported by National Key Laboratory of Mechanical Systems and National Natural Key Product Foundations of China 10231060.

rotation argument that is applied to the optimal solution of the semi-definite relaxation of the problem. Halperin and Zwith [8] further improved the approximation ratio to 0.7016 by strengthening to SDP relaxation with the triangle inequalities. All these algorithms are based on the semi-definite relaxation of problem MB

$$(SDP): \begin{cases} SDP(S) = \text{Max} \ \ L \cdot X, \\ s.t. \ \ Diag(X) = e, \\ \ \ ee^T \cdot X = 0, \\ \ \ X \succeq 0, \end{cases}$$

here $L = \frac{1}{4}(\text{Diag}(We) - W)$, $X \in R^{n \times n}$ is a symmetric matrix, $L \cdot X = trace(LX)$ is the matrix inner product, and $X \succeq 0$ means X positive semi-definite. It is clear that (SDP) is a relaxation of (MB), and since $X = xx^T$ is feasible for (SDP) for any feasible solution x of (MB), we have $SDP(S) \geq MB(S)$.

In this paper, we will propose a continuous model for the solution of max-bisection problems, and a feasible direction algorithm without line search to solve the resulting continuous model. Unlike the available relaxation methods for max-bisection problems, NCP functions are employed to convert the max-bisection problem to a continuous nonlinear programming, and then the resulting nonlinear programming problem is solved using the feasible direction method without line search. The convergence property of the proposed algorithm is studied, and numerical experiments and comparisons on some well-known test problems and on some randomly generated problems are made to show the efficiency of the proposed algorithm on both the CPU times and solutions.

The rest of paper is organized as follows. In section 2 we convert the max-bisection into a continuous nonlinear programming problem by using NCP functions. The relationship between the solutions of the max-bisection problem and the resulting nonlinear programming problem is analyzed. The feasible direction method without line searches are presented in section 3. The convergence of the algorithm to KKT point of the resulting nonlinear programming is proved. Numerical results and comparisons are reported in section 4, and it is observed that the algorithm is effective and efficient on both the CPU times and the solutions. Section 5 gives the conclusions.

2. The Continuous Model of Max-Bisection Problem

In this section we formulate the max-bisection problem into a continuous nonlinear programming by using NCP functions, and analyze the relationship between the solutions of the max-bisection and the resulting continuous nonlinear programming.

The max-bisection problem can be rearranged as

$$(MB): \begin{cases} MB(S) = \text{Max} \quad x^T LX \\ s.t. \ \ e^T x = 0, \\ \ \ x_j^2 = 1, \quad j = 1, \cdots, n, \end{cases}$$

where $L = \frac{1}{4}(\text{Diag}(We) - W)$. If $w_{ij} \geq 0$ for all $i, \ j$, then L is a Laplace matrix, and hence, positive semi-definite ($L \succeq 0$) [9]. Without loss of generality, we will assume, in the rest of the paper, that L is positive definite and $L_{ii} > 0, i = 1, \cdots, n$. because of no any effects on optimal solutions.

Adding constraints $-1 \leq x_j \leq 1, \quad j = 1, \cdots, n$ to problem (MB) gives no effects on its